Network Working Group	W. Ivancic
Internet-Draft	NASA GRC
Intended status: Experimental	W. Eddy
Expires: January 7, 2013	MTI Systems
	A. Hylton
	D. Iannicca
	J. Ishac
	NASA GRC
	July 6, 2012

TOC

Store, Carry and Forward Problem Statement draft-ivancic-scf-problem-statement-00

Abstract

This document provides a problem statement for non-realtime communication between systems that are generally disconnected, requiring multiple network hops between source and destination, that may never be fully connected end-to-end at any given time. This document describes a number of use cases that motivate having a standard method to communicate between such systems, as multi-organization and multi-vendor support and interoperability is highly desirable. These include dismounted soldiers, sensorwebs, medical devices, animal tracking, low-earth-orbiting satellites and data mule scenarios. To avoid confusion in terminology when trying to focus on the problem and requirements without bias towards particular technical solutions, at this time, we refer to the protocol instances that would support such communications as Store, Carry, and Forwarding (SCF) agents, and refer to their activity as SCF networking. The concepts involved in SCF networking are not entirely new and several facets of the problem have been solved in multiple different ways. This document describes the core SCF problem and gives an assessment of the ability to use existing technologies as solutions.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, except to format it for publication as an RFC and to translate it into languages other than English.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as “work in progress.”

This Internet-Draft will expire on January 7, 2013.

Copyright Notice

Table of Contents

TOC

1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 **[RFC2119]**.

"What's in a Word. Words make a difference. They affect how we think about something. The terms chosen to describe a concept are a crucial part of any model. The right concepts with terms that give the wrong connotation can make a problem much more difficult. The right terms can make it much easier. Adopting the mindset of the terms may allow you to see things you might not otherwise see." - John Day **[Patterns]**

In developing this document, we have intentionally avoided some terminology used by other protocols - particularly store-and-forward protocols - to avoid biases and confusion that may otherwise ensue.

- Container - the application/user data to be transported over the network as well as a checksum of that information. Containers may include sub-containers.
- Container Aggregation - The process of organizing one or multiple containers as sub-containers inside another larger container.
- Container Deaggregation - The process of removing one or more sub-containers from a larger container. This differs from fragmentation because rather than creating new containers, deaggregation operates on existing sub-containers.
- Container Fragmentation - The process of dividing a single container's contents into multiple new containers which will need to be eventually reassembled back into the original container before delivery to the application.
- Container Reassembly - The process of recombining the contents of multiple containers into a single container that they were originally part of, and that needs to be delivered to the application intact.
- Delay - propagation delay between SCF agents. Delay does not include disconnection time.
- Disruption - a relatively short period of disconnection within an otherwise well-connected network (e.g. a loss of connectivity in the range of seconds to perhaps minutes)
- Disconnection - a relatively long period where communication between a significant proportion of hosts is not possible for various reasons (e.g. due to the inability to close a radio link)
- Metadata - synonymous with a Container's Label

- SCF - Store, Carry and Forward
- SF - Store-and-Forward, or "store and forward" as used generically in other literature (where the presence of hyphenation varies)
- SCF Agent - a protocol instance providing SCF services to an upper-layer user/application
- Shipping Label - metadata describing the characteristics of a container and its forwarding requirements
- Sub-Container - A small container residing inside a larger container.
- Transport Capacity - (as a first order approximation) the combination of bandwidth and contract time.

TOC

2. Introduction and Background

Internet technology has become pervasive and is now present in many types of devices that end up being deployed in the field for use in scenarios where they do not have good (or any) actual Internet connectivity. The networking stacks are used to support data transfer during episodes of connectivity and avoid reliance on many typical infrastructure services (e.g. DNS). For instance these devices may be only intermittently connected to other devices, and are used to support data flows where the source and ultimate destination might never fully connected to one another at the same time. These applications operate highly asynchronously with non-realtime constraints on their communication. Often there are intermediate relaying nodes (or agents) that must "carry" the data while waiting for connectivity to develop. The means for relaying data has been highly specialized in such systems (almost per-deployment), and varies widely, with little code-reuse or commonality in the supporting network design. This "problem statement" document describes several of these scenarios generically, motivates the development of a common solution, and describes shortcomings in existing technologies.

This problem statement is explicitly not trying to look at the situation were a smart phone or mobile computer is temporarily off or removed from the Internet and then is reattached directly to the edge of a well-connected network. Such systems are well-suited to utilize standard Internet protocols and are able to support realtime communications when connected. The systems and applications that this document is concerned with are primarily operating with a much higher level of asynchronism between the data producers, individual relays, and eventual consumers. We call these "Store, Carry, and Forward" (SCF) systems to distinguish them from typical Store-and-Forward (SF) systems which operate over a better connected infrastructure. This section clarifies the distinction between SCF and the better-understood SF concept, which is implemented by a number of different networking technologies.

To understand SCF systems, it is useful to look at (very) early communications. Relay systems have been around since the beginning of civilization. In ancient times rulers utilized intelligence gathering via courier services to convey and obtain information. Relays consisted of runners, messengers, and even pigeons conveying messages and documents. These types of relay agents had only intermittent connectivity with one another and needed to hold onto messages for possibly long amounts of time before delivering them. Later relay systems included the ancient Greeks using fires and semaphores and in the 18th century, and the French using a system of telescopes and semaphores. These involved relatively well-connected systems of relay infrastructure, compared to the earlier methods that involved physical carriage of the stored messages for some time in order to reach the next forwarding point. Telegraph and later systems had equally well-connected infrastructures.

In computer networking, numerous technologies that support SF message communications between systems have evolved, and some have incorporated pieces of what SCF systems require. We very roughly group these developments into "generations" in order to highlight a general progression of capabilities. This is not prescriptive, and though some detailed aspects of the classification may be debatable, the basic notions hold.

1st generation Store and Forward systems consisted of Message switching, with buffering of messages at intermediate nodes in order to handle intermittent connectivity. There was little or no automation, intelligence, or capabilities for forming routing tables, security, network management, and handling anything but rather slow-scale dynamics. Examples include **UUCP**, **FidoNet**. "In FidoNet As all modem phone numbers are published in the nodelist, point-to-point transfers are always possible. But, as store-and-forward capabilities are specified in the basic standards, email tends to be routed through a world-wide hierarchic topology and enews via a world-wide ad hoc, but generally geographically hierarchic, acyclic

graph."

2nd generation SF consist of Internet email via **SMTP** + **POP** / **IMAP** + **S/MIME** + etc. Key Features include separation of message transfer agents, user agents, and message submission/delivery agents. There are increased capabilities for security and management. There is some (weak) separation between message format and message transfer protocols. Email servers generally operate within a well-connected environment. There are major performance problems outside well-connected environment because there is little diversity in message transport between nodes, and little or no improvement in dealing with dynamics.

3rd generation SF protocols are an advancement over 2nd generation concepts. They are used to implement messaging middleware and applied as the basis of enterprise service bus systems. There is an increased separation between message format and message transfer protocols with increased message transform / mediation capability. There has been a proliferation of proprietary formats, APIs and systems, with great diversity in capabilities. Generally, there are still problems operating outside a well-connected environment, and the security mechanisms are not advanced beyond 2nd generation ones. Examples include: **JMS**, **AMQP**, **STOMP**, and **XMPP**.

4th generation SF protocols are directed at systems with long delays and intermittent connectivity and are known as Delay Tolerant Networking (DTN) RFC 4838 **[RFC4838]**, RFC 5050 **[RFC5050]**. There is very strong separation between format and transfer protocol as well as strong separation between format and addressing/routing. Architecturally, there are many alternatives available for transfer, addressing, and routing with heavy tailoring per each pocket of deployment. Security is possible for limited subset of intended use cases. There are a number of experimental implementations including Interplanetary Overlay Network (ION), DTN2 and others including substantial profiles of features and capabilities. **http://www.dtnrg.org/wiki/Code**.

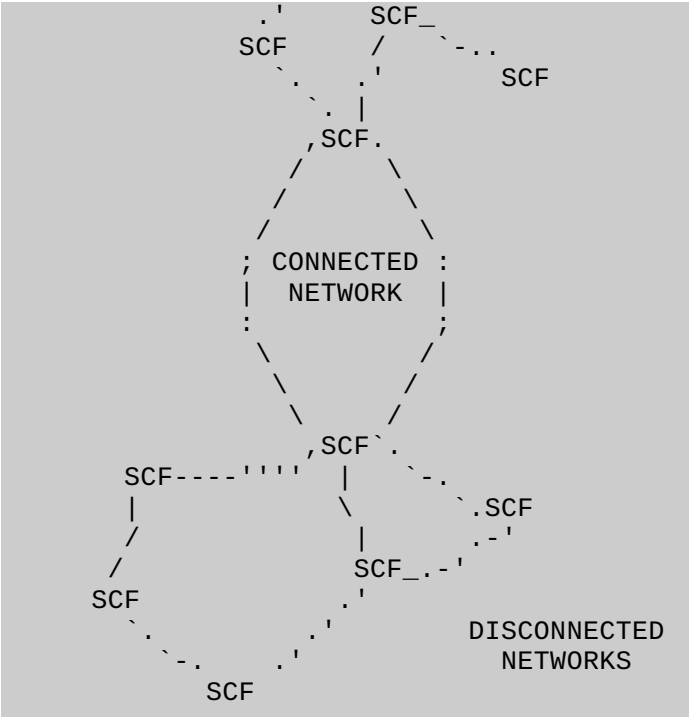
5th generation (conceptual) systems include significant amounts of longer-term storage that can be "carried" between episodes of connectivity as a main component (i.e. Store, Carry and Forward) There is an increased separation of message metadata (i.e. labels) from message body (i.e. containers) beyond the 4th generation, enabling new approaches to security, forwarding, and possibilities for pull-based routing. Emphasis is on the "carry" function and need for strong automation in management of stored data, including support for implementing policy in QoS, security, and routing. 5th generation systems that embody the SCF concept have not yet been widely deployed. The fielded applications that could benefit from such SCF capabilities are either using point-solutions adapted from prior-generations of SF technology, or are lacking the strength in automation, security, and other features that the SCF technology should provide. Later sections of this document describe a generic network architecture expected to be supported by SCF / 5th generation systems, the envisioned scenarios and use cases for these systems, more detailed comparison/contrast with existing / prior-generation systems, and a summary of lessons learned from experience with earlier systems.

---

3. Generic Architecture

TOC

Figure 1 illustrates a generic SCF network architecture, with the SCF agents (labelled "SCF") frequently partitioned into time-varying disconnected subsets. Depending on specifics of an individual scenario, it may be likely that some SCF agents are permanently attached to a connected network in order to provide stable gateways to/from the other SCF agents. However, in general, the system should be considered to consist of a number of primarily disconnected SCF agents at any point in time. The importance of this consideration as it relates to design, implementation, and test of potential SCF protocols is emphasized later in this document, as it has plagued prior SF systems.



Store, Carry and Forward Network

Figure 1

When visualizing a SCF network, it may help to think more along the lines or a topologically dynamically-changing (mobile) relay system of agents that periodically can communicate with one another, and may be able to make at least rough predictions about their future contacts. The distribution of news and mail in the mid-1800s as described in Herman Melville's book, "Moby Dick," is a good analogy.

"For the long absent ship, the outward-bounder, perhaps, has letters on board; at any rate, she will be sure to let her have some papers of a date a year or two later than the last one on her blurred and thumb-worn files. And in return for that courtesy, the outward-bound ship would receive the latest whaling intelligence from the cruising-ground to which she may be destined, a thing of the utmost importance to her. And in degree, all this will hold true concerning whaling vessels crossing each other's track on the cruising-ground itself, even though they are equally long absent from home. For one of them may have received a transfer of letters from some third, and now far remote vessel; and some of those letters may be for the people of the ship she now meets.....

...Every whale-ship takes out a goodly number of letters for various ships, whose delivery to the persons to whom they may be addressed, depends upon the mere chance of encountering them in the four oceans. Thus, most letters never reach their mark; and many are only received after attaining an age of two or three years or more."

Another analogy that illustrates aggregation and deaggregation are the parcel post delivery companies. Here, individual packages (containers) are delivered from a source to destinations via numerous transport mechanisms (e.g. trucks, planes, trains and boats). Along the way, these packages are aggregated into larger and larger shipping containers as they move further from the source and then and deaggregated into smaller and smaller containers as they move closer to the destination. Such aggregation and deaggregation enable scaling of the system. There is a strong parallel between this flow of packages and the data flows seen in some of the scenarios described later in this document.

4. Operational Considerations

TOC

Some of the key operational considerations for SCF are:

- What types of applications might be suitable to utilize SCF networking?
    - Engineering Telemetry - Accumulated over time for offline

monitoring and analysis of some device or system's performance, which may be related to long-term administration of the device, but occurs in non-realtime and at a remote location. Fidelity of the received data is important, though partial delivery of data may be acceptable and more desirable than slower delivery of complete and fully accurate data. It is expected that a telemetry-sending application may operate in a fire-and-forget mode, where it does not retain data after sending.
    - Science Data Gathering - Similar to engineering telemetry, but sensor data is collected at a potentially much larger volume or over a much longer timescale. Accuracy of the delivered data is critical, and timeliness in routing may be sacrificed to provide a complete and error-free data set. Due to the size of data sets collected, having multiple copies in-flight within the network may be undesirable, and end-nodes may need to purge old data after it has been sent in order to gather new data.
    - Software Update - Numerous deployed devices that may never be able to contact an update server in realtime may need to have patches or updates deployed and activated. This can require high reliability and guarantee of eventual delivery of the data, even if the latency involved in applying the update is not extremely critical. The sender is likely to retain access to the sent update/patch perpetually, even after copies have been distributed into the network. While some acknowledgement of reception end-to-end may be desirable, this might be inferred through other means at the application level (e.g. via telemetry) rather than requiring SCF-level acknowledgement.
- In general, any distributed application where senders and receivers can operate asynchronously in non-realtime, without any real-time requirement on the infrastructure (e.g. to do resolution of DNS names) might be able to function over an SCF service.
- What are the potential deployment environments and platform capabilities?
    - Some relevant use cases are discussed in detail in the following section. In general, the SCF agents may be either co-located or independent of the hardware/software platforms that host the end-applications. Aside from having a non-trivial amount of persistent storage, very few assumptions can be made about the SCF agent computing platforms. Typically, they will have to be embedded systems, e.g. within a device that's part of some other portable electronic system (e.g. handheld device, medical implant, avionics hardware, etc.) rather than typical workstations and servers. This means that links are expected to be (much) less capable and more time-varying than wired Ethernet, and frequent administrative access is not likely to be possible.
- What are the upper layer user/application data set sizes?
    - From existing systems in-use that could benefit from SCF, at least several GB of data collected onto an SCF agent between contacts with other SCF agents is possible. There are also applications where only several kilobytes of container are necessary.
- What are the traffic patterns?
    - In envisioned SCF scenarios, movement is not fully random, even for mobile ad hoc networks, though at the very edges, it may appear random.
    - In envisioned SCF scenarios, information flow is not fully random, even for mobile ad hoc networks.
- What type of interface between SCF agents and end applications is feasible?
    - Applications should be able to select their own globally unique identifiers and notify SCF agents of them, along with providing proof of ownership. SCF agents may be able to notify applications of pending received data, but applications are always able to poll a SCF agent for such data as well.
- What interaction between SCF agents is expected?

- When in contact with one another, SCF agents minimally need to be able to identify one another securely and prove that they can be trusted as relays for a given destination application. Agents should be able to indicate (or deny) forwarding of individual containers, based on exchanging their labels only.

---

5. Use Cases and Deployment Scenarios

TOC

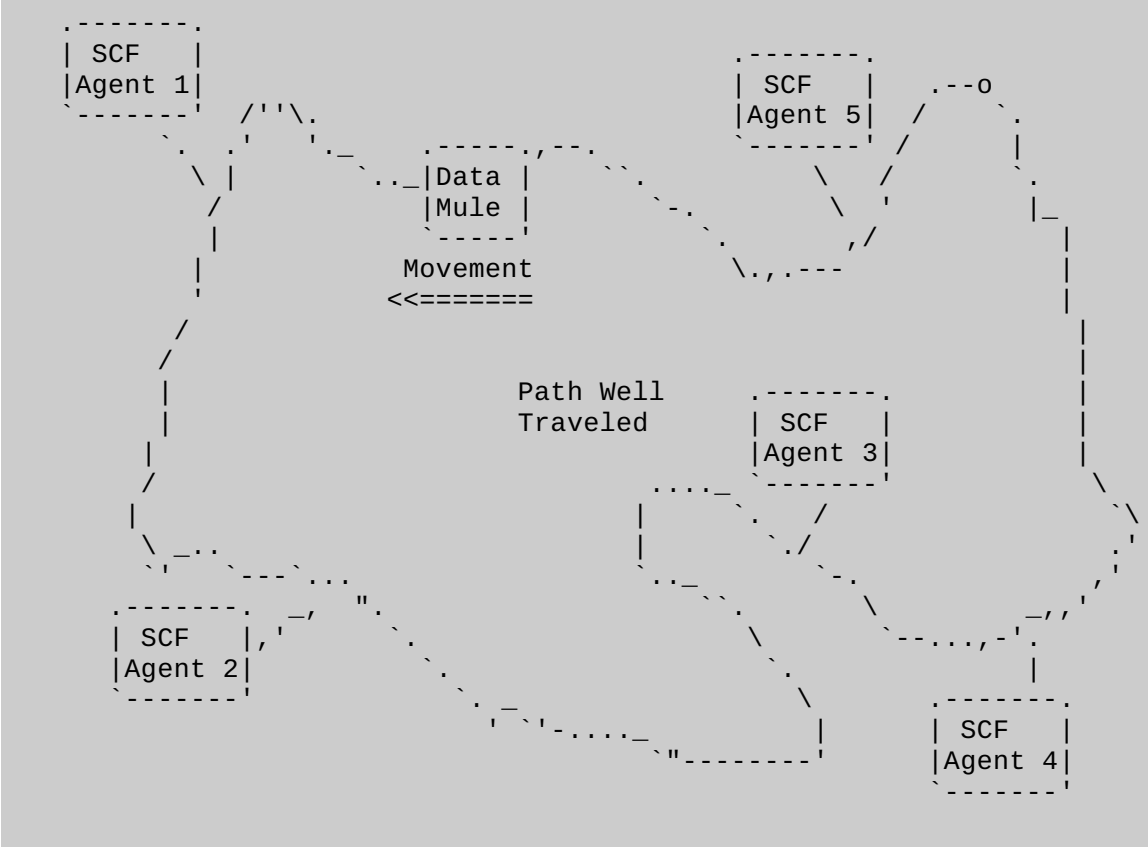
There are numerous deployment scenarios for SCF systems. The following section highlights a few more common scenarios.

5.1. Data Mule

TOC

The Data Mule scenario is a common generic scenario and shows up many more specific types of scenario. In the Data Mule scenario, SCF agents communicate with each other mainly via some type of circulating entity carrying data, called the "mule". This entity may be a unmanned (or manned aircraft), a ship, a bus, or any type of vehicle that periodically moves over the same relative area. Connectivity is likely to consist of high periods of disruption followed by short periods of connectivity over relatively high-bandwidth, low-delay, and possibly symmetric links.

In the Data Mule scenario, connectivity is generally of the episodic variety (opportunistic). There may be one or a larger number of mules; each of which runs its own SCF agent.



Data Mule Scenario

Figure 2

Within this type of Data Mule scenario, the generic use case for SCF networking involves an

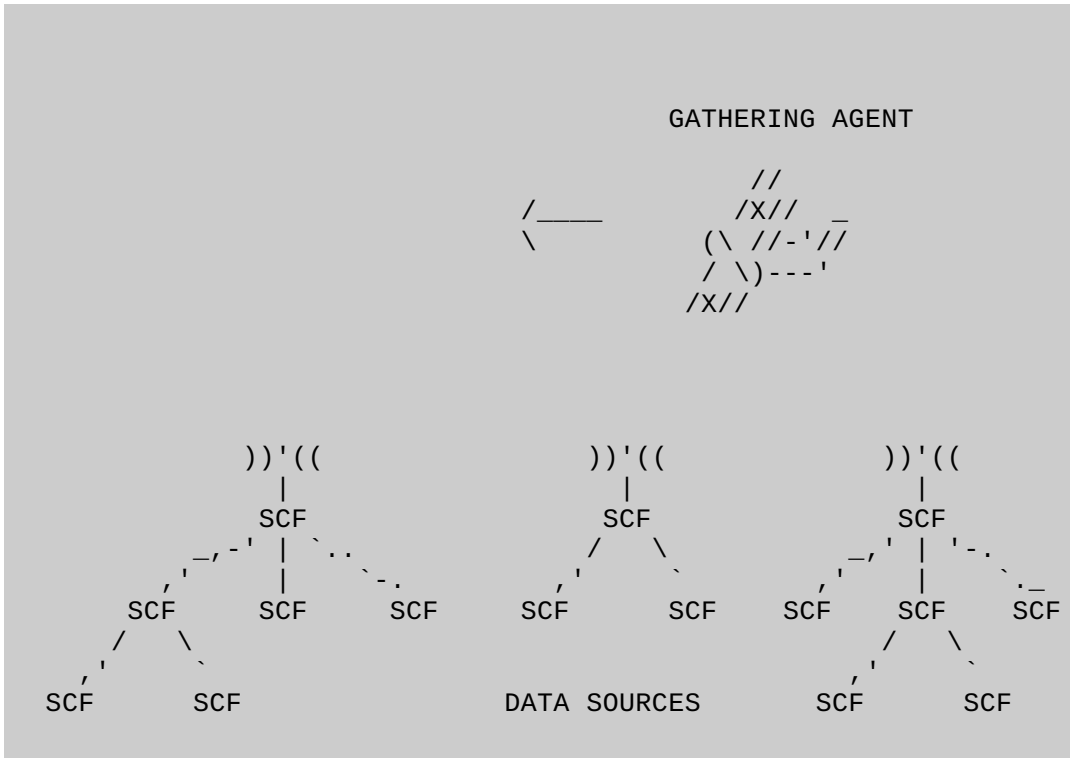
application being able to push its data into containers on a SCF agent, who then interacts with the Data Mule SCF agents in order to deliver the containers to destination applications attaching to other SCF agents. In order to realize this use case, the SCF agents need to be identifiable to one another during periods of episodic connectivity, and somehow the mule needs to be able to express its expected future capability to relay containers towards the destination application.

Data Mule is a common military scenario. It is often used to join partitioned connected networks such as groups of mobile ad hoc networks (manets). In figure 1, the SCF Agents could be concentrator points in a manet cluster that enables communication with disjoint manets on the battlefield thus enable communications between clusters on the far ends of the communication infrastructure, the edge networks.

TOC

5.2. Data Gathering

The generic Data Gathering scenario is also quite common and applicable to SCF networks. Specific use cases involving sensorwebs, medical monitoring and animal tracking would fit into this scenario. Figure 3 illustrates a sensorweb where some sensors wake up and forward data through other SCF agents until they reach an SCF connected to a more powerful radio system. A gathering agent may then come by from time to time (e.g. days, weeks, months)and vacuum up the data.



Data Gathering Scenario

Figure 3

Major challenges of the use cases in a Data Gathering scenario that go beyond those of a Data Mule, are related to the increased level of complexity in the topology between SCF agents. There is potentially less predictability, potentially more heterogeneity (or hierarchy) in SCF agent capabilities, and potentially a higher risk of routing loops or wasted resources.

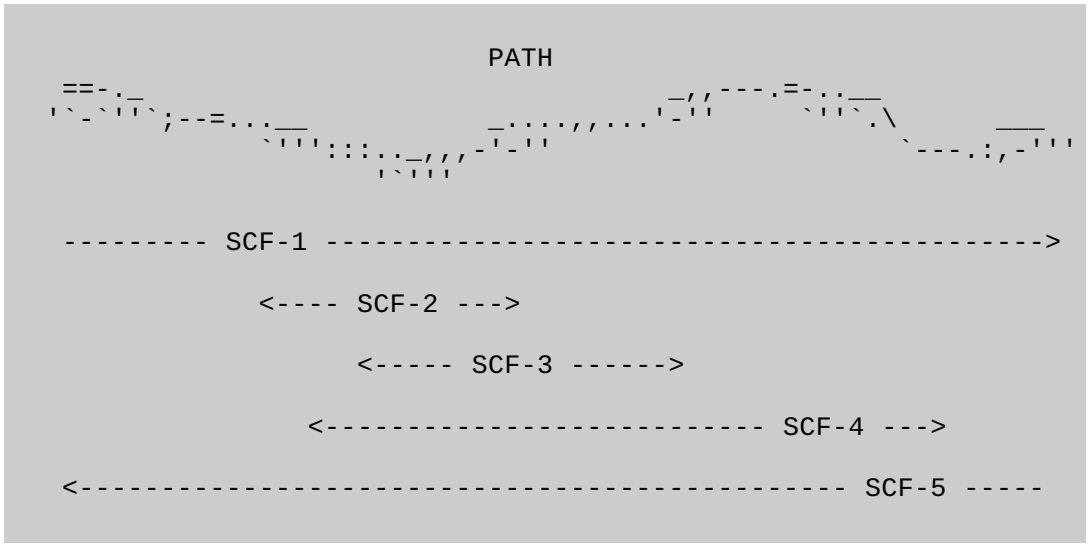
The use cases for Data Gathering do however involve data flows that are generally either all directed from sensors up through the Gathering Agents or down from the Gathering Agents, so these still represent a sort of core network that all containers eventually go through (similar to the Data Mules).

5.3. Traveling the Beaten Path

TOC

There are many instances where communications may occur between entities traveling a well worn path. In this scenario, communication is more ad hoc than the data mule example. The probability of encountering other SCF agents is quite good. Such scenarios include: communication in mining operations, among hikers, among boats along well traveled waterways, within the fisheries industry (the Moby Dick example) and along trade routes.

Figure 4 illustrates Traveling the Beaten Path. Consider a nomadic trade route in a third-world country. Here, SCF-1 may travel from the one end of the path to the other in one direction while SCF-5 moves in the other direction. SCF-1 and SCF-5 will encounter all other SCFs along the way. SCFs 2, 3 and 4 only move along portions of this trade route. Most likely, none of this information is known in advance and the movements may or may not be predictably repeatable.



Traveling the Beaten Path

Figure 4

5.4. Rapid Disruption

TOC

Many wireless networks - particularly military wireless networks are episodically connected because of terrain, foliage, weather, jamming, or desire to evade detection. Furthermore, radio signal power fades can cause rapid very-short periods of disconnection. All of these instances may result short periods of disruption as well as rapid changes in topology.

When connected, however, the connectivity may be relatively complete to a large number of other nodes in near real-time.

SCF provides some potential solutions for such networks. SCF routing may be capable of moving containers towards destinations via store and forward if the proper naming structures, addressing and routing algorithms can be developed.

5.5. Dismounted Soldier

TOC

On the battlefield, it often occurs that a group of soldiers is on a mission and arrives via a vehicle or group of vehicles, one which may have very good connectivity to larger networks. Once dismounted, much of the communications may be via use of the vehicle

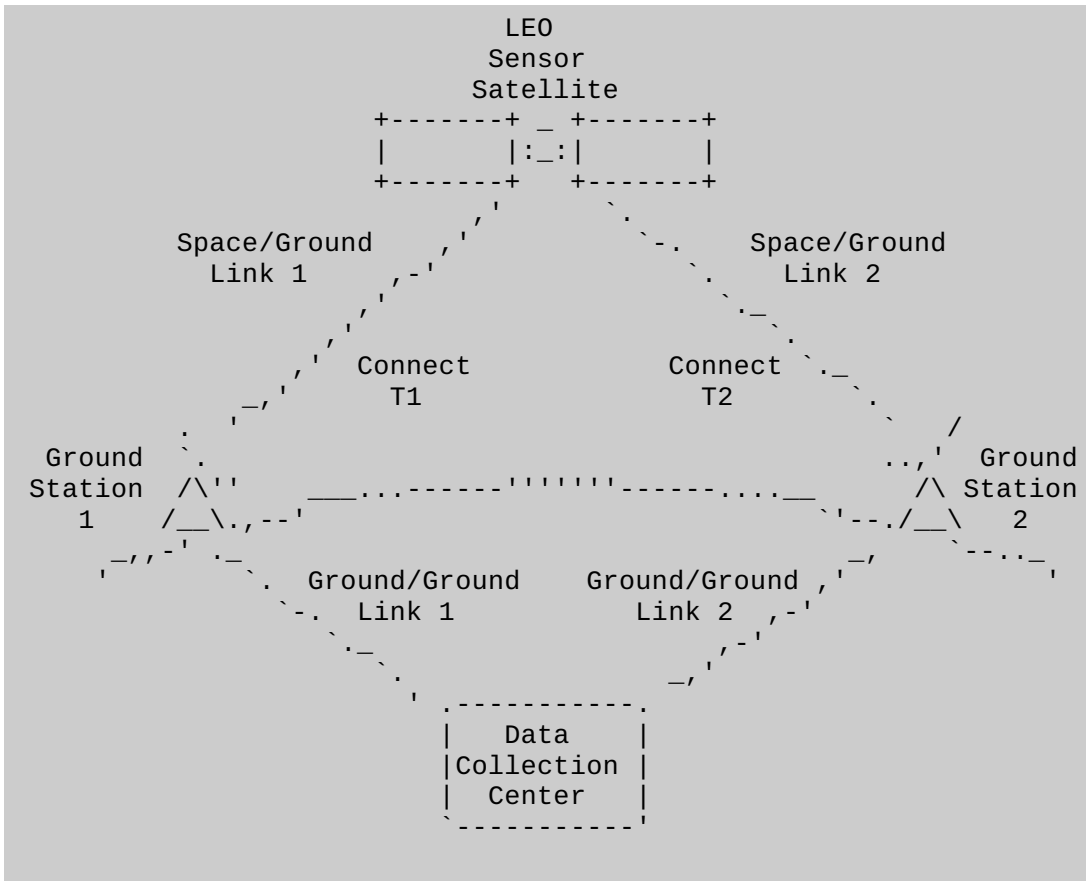
communication systems as a relay or anchor. Once the group moves a significant distance from and from each other, they become disconnected for some period of time. At this point, SCF becomes necessary to improve communications and maintain situational awareness. SCF provides this communication in two ways. One can communicate via multi-hops through other SCF agents and/or one can offload data to the network once within radio contact of the anchor relay on the vehicle.

This scenario is also applicable to first responders during disaster situations where infrastructure may be severely damaged.

TOC

5.6. Low Earth Orbiting Sensor Satellite

The Low Earth Orbit (LEO) scenario described is for a sensor satellite communicating directly with ground terminals. One such network is described in reference **[UK-DMC]**. Note, in this scenario, no geostationary relay satellite is involved. Here, the contact times may be known in order to direct the satellite transmitter to turn on. Some type of automated hailing could also be used.



Low Earth Orbit Sensor Satellite

Figure 5

Low Earth Orbit (LEO) is a low-propagation-delay environment of less than ten milliseconds delay to ground, with long periods of disconnection between passes over ground stations. Contact times consist of a few minutes per ground station for Earth satellites orbiting at a couple hundred kilometers altitude (~100 minutes per pass). Thus, the more ground system that are available, the greater the potential for contact. The ground stations are connected across the terrestrial Internet, which has different operating conditions (congestion-sensitive, always on) from the private links between satellite and ground station (intermittent but scheduled, and dedicated to downloading with no need for congestion control.)

In this scenario, the SCF agent onboard the satellite does not need to perform forwarding of

received bundles. The satellite is a source for sensor data and may be a sink for command data.

The main reason to use SCF in this scenario is to provide a standardized store-and-forward technique and to break the control loops between the space/ground link and the ground/ground link.

There are numerous companies and systems today that transfer extremely large sensor data sets from LEO to ground without a standardized SCF method. Those data sets are in the multi-gigabyte region and growing.

The space/ground link is a private link that is congestion-free. The desire is to fully utilize that link when it becomes available. For sensor satellites, the space/ground link may be highly asymmetric, with command uplink to the satellite at kilobit-per-second rates and data downlink from the satellite at tens to hundreds of megabits per second. Transport protocols that operate well in this highly asymmetric environment are required.

The ground/ground link is over the Internet where the data rate may not be controllable and may not be known. Furthermore, congestion-friendly transport protocols are required.

TOC

6. Consideration of Existing Technologies

In this document, we characterized DTN-based systems as 4th-generation SF rather than SCF systems. Several aspects of DTN are highly desirable for SCF though, and DTN technology may represent a basis for developing SCF standards. DTN utilizes "bundle agents" that are similar to the "SCF agent". Several DTN routing protocols exist at varying levels of maturity that can work well for individual SCF scenarios that have been outlined. For instance, Contact Graph Routing is particularly useful in scenarios where future connectivity is predictable/known ahead of time.

The SCF container aggregation and deaggregation bears some surface similarity to bundle fragmentation operations in DTN, but there are major differences. SCF containers are intended to be aggregatable within the network, even if they are not portions of the same original container from the application. Additionally, some SCF applications (e.g. science data collection) may find (optional) partial reception of subsets of large containers that have been deaggregated into smaller containers, to still be useful, whereas DTN only delivers entire (reassembled) bundles. This does require the data formats used by such SCF applications to be self-synchronizing, so that they can be parsed, but this is an issue for the application.

SCF scenarios require some features that are not yet a part of the DTN specifications:

> The ability to avoid DoS by propagating an application's permit/deny filters to SCF agents.
>
> The ability to generate and prove ownership of globally unique application identifiers.

DTN goes beyond SCF in one way, which is the targeting of operation over very high latency data links. SCF does not explicitly attempt to operate over such links, though it may end up being possible. Since these links are mostly only applicable to deep-space scenarios with small numbers of nodes, SCF motivations do not include high-delay.

This document also identified JMS message-broker systems as 3rd-generation SF rather than SCF systems. JMS "messages" transferred between "brokers" and applications are similar to the containers transferred between SCF agents and applications. JMS offers both point-to-point (unicast) and publish-subscribe (multicast) models of communication. JMS uses named "queues" (in the point-to-point model) or "topics" (in the publish-subscribe model) in order to identify destinations. JMS brokers often implement a "durable" messaging service that allows messages (and queues) to persist even when the applications that created them (or need to receive them) are disconnected from the broker.

SCF scenarios require some features that are not yet really reflected within the JMS specifications:

> Multi-hop relaying among brokers and secure propagation of information about the queues/topics present or acceptable is not standardized.

Communication of an application's desired permit/deny filters on queues it owns is not standardized.

JMS is an API and not a protocol standard. This is the primary hurdle in using JMS to support SCF; as the wire-protocols and other mechanisms used in a particular JMS implementation are not necessarily compatible with others. SCF requires full vendor/platform interoperability in order to be cost-effective to pursue instead of point-solutions. JSM is also significantly focused on transfer of Java objects rather than generic containers of bytes as SCF should be.

One of the biggest challenges to using existing systems (whether they be DTN implementations, JMS products, or some others) is that most have been designed to include a multitude of additional (optional) features and this results in, at best, limited compatibility between implementations. For instance, the DTN Bundle Protocol is an excellent platform for experimentation due to its flexibility and ease of defining new "blocks" to implement different functionalities. DTN has been used or demonstrated in a wide range of scenarios with differing needs, including simulated military exercises, connecting people in remote regions, moving data from LEO spacecraft, deep-space missions, mining operations, and others. However, individual implementations have grown up to support distinct subsets of the defined blocks, identifier schemes, and algorithms that suit the unique properties of their pet environments. Developing, and then maintaining, a baseline for compatibility has not been a primary concern. For an operational system, a baseline profile of the required functionality would need to exist, which could be present across the spectrum of vendors. For SCF, this type of profile is not present in the existing systems to a level that would enable the scenarios described in this document. Saving energy and running on very small devices (e.g. sensors and embedded medical devices) also motivates having such a profile that could aid in developing very small, yet fully compatible, implementations.

MQ Telemetry Transport **(MQTT)** is a lightweight network protocol used for publish/subscribe messaging between devices. MQTT was designed for low-bandwidth, high latency networks while attempting to ensure reliability of delivery. A major design criteria was simplicity - must be simple to implement and must not add too many “bells and whistles” while providing a solid building block which can easily be integrated into other solutions. MQTT is designed to handle frequent short periods of network disruption using a technique called "Last Will and Testament". Although not part of the specification, MQTT has been modified to operate in multi-hop environments.

TOC

7. Characteristics of Information

Since information has to be transported and stored, it is important to look at some of the key characteristics of the information being acted upon in a SCF. All information has a source and destination. All information has a size component. The size may be very small or quite large, bytes to GBytes. Size is important because it takes up storage and because transmission bandwidth and contact times limited the amount of data that can be sent during any given contact time.

Information may have security restrictions place on it - sensitive or restricted (for your eyes only). However, this can be handled at the application layer such as is done by securing email.

All information has a useful lifetime. It may be very short (seconds) or very long (days, weeks, years). Regardless, it is only the users of the information that know what the real useful lifetime is and it is the application that would be required to set that lifetime. With the exception of specific cases, it is not at all clear that the application can generally make that decision.

When investigating the use of bundle lifetimes in DTN deployments and implementations one finds that the lifetimes have generally been set to match the duration of the experiment. There are instances where some finer granularity has been deployed such and in the Defense Advanced Research Project Agency's (DARPA) Wireless Network after Next (WNaN) where lifetimes of minutes and hours are used depending on the data. Upon further investigation, one finds that the lifetime is used for two purposes: expedited forwarding and purging stale information from the system. Thus, the real requirement is that one should be able to expedite forwarding of priority containers and purge stale containers from the system. There may be other means to accommodate this requirement without having to burden the SCF agents with the management of "time" - particularly per container.

Often data has a "freshness" characteristic. For a given application data that is more recent (fresher) is often of greater value that data that is older (stale). In such cases, it may be more important to forward the most recent data rather that the data that is near its useful lifetime. One might even purge the system of stale data. One example that illustrates why data freshness is important would be reception of stock quotes. Obviously, one would not expect an SCF systems to be used for commodities trading due to disruption and ordering issues (assured timeliness). Rather, applications such as sensor data transmissions, software updates or distributed security-key databases are more amenable to SCF deployments.

TOC

8. Network Management

Network management to keep the network running smoothly. It is required for system configuration and maintenance and monitors the system to determine faults, performance, security issues and accounting. From the scenarios presented, it appears that network management is likely to be per scenario and may be effectively accomplished out-of-band. For example: in the Data Mule scenario one may manage the data mule, but not the edge SCF systems. In the Data Gathering scenario one is likely to preconfigure the remote sensor nodes and only manage the data gathering SCF and perhaps the data concentrator SCFs, the ones with high-power radios.

This does not imply the network management could not or should not be performed in-band. Only that it may not be required.

Since resources (e.g.bandwidth, transmit power, and storage) are a precious commodity in SCF networks, policy that manages those resources is expect to be a major component of system configuration. For example: a particular SCF agent may restrict particular information sources to limited storage space and limited storage time. Such policy may restrict all information to a limited storage time in order to purge stale containers. Also, particular sources may get preferential treatment per peering agreements.

TOC

9. Lessons Learned Summary

There are numerous lessons to be learned form previous deployments of manets and 4th generation store and forward network such as DTNs. Some of the more critical and important pieces of knowledge are listed below:

- SCF systems are generally connected via radio networks. Some radio systems may take far less power to listen than to transmit, though this varies by individual link technology. Wasted transmission is wasted power on a wireless system and can quickly drain a battery. The problem is compounded for devices whose entire lifetime is determined by their battery (e.g. non-rechargeable sensor nodes). Thus, reducing wasted transmissions is high desirable.
    - The ability to reactively fragment large data sets en-route is highly desirable. This has been demonstrated in DTN experiments.
    - Routing loops in the SCF will not be caught by layers below. It is imperative that data dies naturally and quickly so as to not waste bandwidth or transmission power. Such loops have been encountered in early experiments with DTN overlays, and are correctable.
    - It is highly desirable for the sender to know early in a transmission whether or not the receiver will accept the data. This permits a savings in power and optimization of network capacity usage. For instance, in DTN experiments with large bundles, the entire large bundle may be sent, only to be discarded due to security, resource scarcity, or other issues.
- Disconnected networks are difficult, if not impossible, to globally synchronize state across.
- Managing time in a protocol is difficult and adds considerable complexity.
- Having your transport protocol be time-dependent opens is up to security vulnerabilities.
- Having your transport protocol be time-dependent means you cannot use that

protocol to synchronize your system - even for course synchronization.

- It is highly desirable for a receiving agent to determine early within a transfer whether or not to accept the data. Data sets can be quite large utilize significant processing and storage resources for data that may end up being discarded due to security, resource constraints, or other policy issues.
- It is highly desirable to keep forwarding tables small, and make forwarding decisions ahead of time for predicted contacts. Book-keeping type of processing while forwarding a large number of small containers can overload the processing system.
- Testing should be thorough and include exercising both the storage and forwarding systems. Failure to do so will lead to erroneous results. XXX Place I-D.ietf-ivancic-scf-testing here XXX Thus, any testing and validation should exercise both the storage and forwarding mechanisms of the implementation. To do otherwise my lead to misleading results. In addition, when connections occur, it is generally via radio systems.

---

10. Security Considerations

TOC

Applications need to authenticate to a SCF agent before they can send or receive containers.

Authentication of SCF agents to one another needs to be tackled before advertisements of forwarding capability can be acted upon.

Bandwidth, Storage, and Processing Power are precious resources in a SCF. In order to reduce DoS vulnerabilities and properly allocate resources, a SCF should be able to determine whether or not to act on a container based solely on the Shipping Label.

Applications should be able to limit DoS by expressing explicit desires to a serving SCF agent for/against certain traffic selectors. It may be beneficial for this information to propagate between SCF agents, though it should be recognized that any dynamics in these preferences causes a risk of data loss due to lack of synchronization of the filter rules.

While some aspects of Public-Key Infrastructure (PKI) may be applicable to SCF, PKI itself is not because PKI requires connectivity. Public-Keys with caching may be applicable; however, this would require at least some course network synchronization.

---

11. IANA Considerations

TOC

This document neither creates nor updates any registries or codepoints, so there are no IANA Considerations.

---

12. Acknowledgements

TOC

Much work builds on lessons learned from the work performed by the IRTF DTN Research Group.

Work on this document at NASA's Glenn Research Center was funded by the NASA Glenn Research Center Innovation Funds.

Many thanks to Denise Ponchak for aiding in obtaining financial supporting for this activity.

---

13. References

TOC

13.1. Normative References

TOC

[RFC0768] Postel, J., "**User Datagram Protocol**," STD 6, RFC 768, August 1980 (**TXT**).

[RFC2119] **Bradner, S.**, "**Key words for use in RFCs to Indicate Requirement Levels**," BCP 14, RFC 2119, March 1997 (**TXT**, **HTML**, **XML**).

13.2. Informative References

TOC

[Patterns] Day, J., "Patterns In Network Architectures - A Return to Fundamentals," Prentice Hall , 2008.

[RFC4838] Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., and H. Weiss, "**Delay-Tolerant Networking Architecture**," RFC 4838, April 2007 (**TXT**).

[RFC5050] Scott, K. and S. Burleigh, "**Bundle Protocol Specification**," RFC 5050, November 2007 (**TXT**).

[UK-DMC] Wood, L., Ivancic, W., Eddy, W., Stewart, D., Jackson, C., and A. da Silva Curiel, "Use of the Delay-Tolerant Networking Bundle Protocol from Space," 59th International Astronautical Congress, Glasgow Paper IAC-08-B2.3.10 (also NASA-TM-2009-215582), September 2008.

Authors' Addresses

TOC

William Ivancic
NASA Glenn Research Center
21000 Brookpark Road
Cleveland, Ohio 44135
United States
Phone: +1-216-433-3494
Email: **william.d.ivancic@nasa.gov**

Wesley M. Eddy
MTI Systems
Email: **wes@mti-systems.com**

Alan G. Hilton
NASA Glenn Research Center
21000 Brookpark Road
Cleveland, Ohio 44135
United States
Phone: +1-216-433-6045
Email: **alan.g.hylton@nasa.gov**

Dennis C. Iannicca
NASA Glenn Research Center
21000 Brookpark Road
Cleveland, Ohio 44135
United States
Phone: +1-216-433-6493
Email: **dennis.c.iannicca@nasa.gov**

Joseph A. Ishac
NASA Glenn Research Center
21000 Brookpark Road
Cleveland, Ohio 44135
United States
Phone: +1-216-433-6587
Email: **jishac@nasa.gov**